



(11)

EP 3 726 384 A1

(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
21.10.2020 Bulletin 2020/43

(51) Int Cl.:
G06F 11/07 ^(2006.01) **G06F 11/14** ^(2006.01)
G06F 11/16 ^(2006.01) **G06F 11/18** ^(2006.01)
G06F 11/20 ^(2006.01)

(21) Application number: **19170019.4**

(22) Date of filing: **18.04.2019**

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME
Designated Validation States:
KH MA MD TN

(71) Applicant: **Bayerische Motoren Werke Aktiengesellschaft**
80809 München (DE)

(72) Inventor: **Oswald, Florian**
80339 München (DE)

(54) **METHOD AND SYSTEM FOR PRESERVING CONSISTENCY OF STATES DURING FAIL-OPERATIONAL CONTEXT SWITCH**

(57) Provided is a system for updating one or more states relating to a message in a vehicle, the system comprising a first execution unit (104), a state keeping unit (106), and a peripheral unit (102), it is an object of the present invention to improve the reliability of the states. The object of the present invention is achieved by an atom unit (103) and a buffer, wherein the atom unit (103) is configured to carry out the steps: a) receiving a first signal (110) from the first execution unit (104), indicating the message and the states; b) saving the message and the states in the buffer; c) receiving a second signal (120) from the first execution unit (104), requesting the initiation of sending the message; d) sending (125) the message to the peripheral unit (102); and e) sending a third signal (135) to the state keeping unit (106) to update the states.

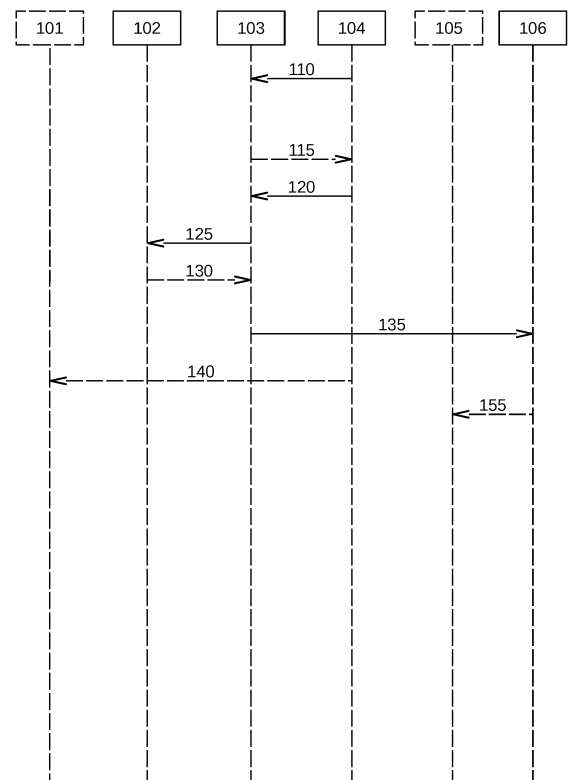


FIG 1

EP 3 726 384 A1

Description

[0001] The invention relates to a method and a system for preserving consistency of states during fail-operational handover and computer-readable medium for the same.

[0002] Autonomous Driving is one of the key E/E (electrical-electronic) architecture push factors in the automotive industry. Progression towards autonomous driving is described in the SAE levels of the according norm SAE J3016 of SAE International. This norm describes six levels extending from level zero to level five, where the latter contains driver less driving or fully automated driving.

[0003] In order for autonomous vehicles to operate safely where no human is required to take over in case of operation failure, the system driving the vehicle must be "fail-operational". The fail-operational requirement reflects the complex and varying environment the vehicle is operating in, and that merely shutting the vehicle off in case of failure (fail-safe in a narrow sense) is in many cases dangerous. Fail-operational can be defined as a system that is designed to be operational even in case of failure of a part of the system. Fail-operational can be used primarily in the context of safety-critical vehicle systems. The definition is however not limited to safety-critical systems, and may also be used in the context of less safety-critical systems, such as navigation systems or vehicle entertainment systems.

[0004] A fail-operational system may perform a hand over from a primary (failing) part, to a secondary (fail-operational) part. A fail-operational part may be fully functional. In this case, the vehicle is fully operational in case of a failure in the primary part but is sensitive to failure due to reduced redundancy.

[0005] In a fail-operational mode (operating on the fail-operational part), the system may also be operational to a lesser extent and/or with less functionality. For safety-critical parts, this should involve functionality which at least ensures the system can safely bring the vehicle to halt at a safe location. An idea behind the "simplex architecture" is that the fail-operational part should be less complex and/or have fewer parts or instruction routines with fewer lines, and therefore less likely to fail. This allows a combination of the functionality of a complex system with the robustness of a simple system.

[0006] Operation of a component in a vehicle may depend on communication with one or more peripheral components such as a CAN communication component. Fail-operational behaviour of such a component may be implemented using a primary component that communicates with the peripheral component, and a fallback component that takes over component operation in case of failure in the primary component.

[0007] The implementation of such a process may not operate correctly, or even fail, when the fail-operational part takes over, since the fail-operational part does not have access to the current states of the vehicle when it is initiated. One alternative to solve this might be to re-

quest states from sensors and/or other units of the vehicle. This is however time-consuming, and may compromise safety during the delay. Alternatively, states could be read from the failing part. This however, may not be reliable due to the low reliability of the failing component. Alternatively, external state storage, common to the primary and the fail-operational part, may be used. There is however a risk that a common external state storage does not have states consistent with messages sent to peripheral devices, since failure may occur in the time gap between the sending messages and the updating of states.

[0008] The deficiency outlined above is solved by the system of claim 1, the method of claim 10, and the computer-readable medium of claims 13 and 14.

[0009] In particular, the deficiency is solved by a system for updating one or more states relating to a message in a vehicle, the system comprising:

- a first execution unit
- an atom unit;
- a buffer;
- a state keeping unit; and
- a peripheral unit,

wherein the atom unit is configured to carry out the steps:

- a) receiving a first signal from the first execution unit, indicating a message and a state;
- b) saving the message and the state in the buffer;
- c) receiving a second signal from the first execution unit, requesting the initiation of sending the message;
- d) sending the message to the peripheral unit;
- e) sending a third signal to the state keeping unit to update the states.

[0010] Advantages of the system include that the system ensures that the states of the state keeping unit is consistent with the message sent to the peripheral unit, and thus with the state of the vehicle, even if the first execution unit encounters a failure. The atom unit will send the message and update the states independent of fault in the first execution unit. The peripheral unit may be a CAN communication component, and the first execution unit may be an electronic control unit (ECU) controlling a function of the vehicle. The states may be an internal representation of the controlling function.

[0011] The vehicle may be operated, at least partially, based on the message sent from the first execution unit, via the atom unit, to the peripheral unit.

[0012] In one embodiment, the atom unit is operational independent of a failure, and/or an error, in the first execution unit.

[0013] Further advantages include that the in case of a fault or an error occurring in the first execution unit, once the atom unit has received the second signal to initiate the sending of messages and storing of states, the atom

unit may continue operation and stores both the message and the state. An additional advantage is that the atom unit may be operational for a redundant component taking over after failure in the first execution unit.

[0014] Independent operation of the atom unit may be implemented by physically or logically separating the atom unit and the first execution unit. As an example, the atom unit and/or the first execution unit may be implemented as using separate logical gate arrays, on the same, or preferably different, hardware. The gate arrays may be realized using application-specific integrated circuit (ASIC) and/or Field-programmable gate arrays (FPGA). The first execution unit may also be implemented using a microprocessor and/or a CPU separated from the execution of the atom unit. The atom unit and the first execution unit may be connected via signals communicated via a bus linking the first execution unit and the atom unit.

[0015] A simple example of a system may include a first execution unit in the form of an ECU controlling the high beam of a vehicle via a CAN communication component (peripheral unit). The ECU may send a signal to an atom unit containing the CAN message to change the head light from low beam to high beam, and a corresponding change of internal states. The atom unit may save the message and the state in a buffer. On receiving a signal to initiate the change, the atom unit may send the signal to the CAN communication component, and update the state in a state keeping unit.

[0016] If the ECU encounters an internal error, consistent states may always be read in the state keeping unit. If the error occurs before the second signal to initiate the message is sent, a low beam will be read. If the error occurs after the second signal is sent, a high beam will be read, corresponding to the actual state of the vehicle.

[0017] The state keeping unit may be locked for reading, such as by the atom unit, during the updating of states, to prevent inconsistent states being read.

[0018] In one embodiment, the failure and error is identified by inconsistency in lockstep execution or voting logic of the first execution unit (104), and/or the failure and error gives rise to an interrupt, preferably aborting the first execution unit operation.

[0019] Advantages of such a system include error detection that is independent of software implementation and/or that will also detect certain hardware defects. By executing instructions in parallel and comparing results (internal memory/registers etc.), an error may be detected in case of inconsistent executions. It may be difficult and/or not reliable on a component level to determine which of two different execution results is the correct one. To address this, a context switch may be initiated to a fallback component and the operation of the first execution unit is aborted. Abortion may include disconnecting the first execution unit from the atom unit and/or the state keeping unit. Other means of determining an error in the first execution unit may also be used.

[0020] In one embodiment, the atom unit is further con-

figured to carry out the step:

b2) sending a fourth signal to the first execution unit, indicating that the message and the state has successfully been saved in the buffer; and

5 wherein the second signal of step c) is sent by the first execution unit in response to receiving the fourth signal.

[0021] An advantage of such a step is that the message to initiate sending the message and storing of the states in step c) may not be performed until the information is correctly stored by the atom unit. The first execution unit may take corrective measures if the information is not stored correctly, such as sending a signal erasing said information and/or re-sending the second signal with the message and the states.

10 **[0022]** In one embodiment, the atom unit is further configured to carry out the step:

b3) checking if the message and the states are correctly saved in the buffer, and

20 sending the fourth signal in response to checking the message and states are correctly saved in the buffer.

[0023] Said step possesses the advantage that a further check on the consistency of the system is performed, such as by reading the saved message and states from the storage and comparing it with the received signal to assess that the information is stored correctly. The atom unit may further send the read information back to the first execution unit so that the first execution unit may compare it with internal information before initiating the second signal of step c) to initiate the sending of the message and the storing of the states.

30 **[0024]** In one embodiment, the atom unit is further configured to carry out the step:

35 f) receiving a fifth signal from the peripheral unit, indicating that the message has been successfully processed; wherein step e) is carried out when the message successfully has been processed.

[0025] This provides improved consistency in case there is an error in the peripheral unit, such as in a CAN communication component or during sending a message on the CAN communication network.

[0026] In one embodiment, the atom unit preferably comprises error correction means, such as voting logic.

40 **[0027]** The advantage of this feature is that a solution is provided in the scenario that the atom unit itself encounters an error during execution, potentially causing the consistency between the states and the message to be compromised. To improve robustness of the execution of the atom unit, the atom unit may rely on voting logic, such as by executing several parallel executions of the same code, preferably an uneven number of parallel operations, such as three, five, or more, and addressing inconsistency between executions by majority voting. The likelihood of two executions failing simultaneously decreases with the number of parallel executions, leading to improved robustness of the system.

45 **[0028]** In one embodiment, the system further compris-

es:

a decision unit; and
a second execution unit;
wherein the decision unit is configured to:
in response to receiving a signal indicating a failure
in the first execution unit, disconnect the first execu-
tion unit from the atom unit, and connect the second
execution unit with the atom unit and/or the state
keeping unit.

[0029] Such features provide a system with improved robustness that, in case of a failure in the first execution unit, may disconnect the first execution unit from the atom unit and continue vehicle operation with the second execution unit connected.

[0030] The vehicle may be operated, at least partially, based on the message sent from the second execution unit, via the atom unit, to the peripheral unit.

[0031] In one embodiment, after the second execution unit is connected with the atom unit and/or the state keeping unit, the second execution unit may read the states from the state keeping unit. The states may be read directly from the state keeping unit, or via the atom unit.

[0032] This has the advantage that the second execution unit may be initiated using a consistent state stored in the state keeping unit. This ensures correct operation of the second execution unit.

[0033] The first execution unit may further be disconnected from the atom unit to ensure that the failing component does not affect the operation of the vehicle, such as by sending messages to peripheral devices, by updating states in the state keeping unit.

[0034] The second execution unit may be connected to the atom unit in the same way as the first execution unit, such that it may perform the sending of messages and states to the atom unit and initiating the sending of the message to peripheral devices and states update via the atom unit. This ensures a consistent state between the messages and the states and allows handover from the second execution unit to the first execution unit to take place at any time.

[0035] After a handover (context switch) to the second execution unit, the first execution unit may initiate a recovery procedure in an effort to return to full operation. If recovery is successful, the decision unit may perform a second context switch back to the first execution unit.

[0036] Since the redundancy of the system may be limited when the second execution unit is operational, measures should be taken to limit the risk of failure in case the second execution unit also fails. The second execution unit may have error corrections measures, such as voting logic, and may implement a reduced set of functionality to limit the risk of failure.

[0037] In particular, the object is solved by a computer-implemented method comprising steps:

a) receiving a first signal indicating a message and

a state;
b) saving the message and the state;
c) in response to saving the message and the state, receiving a second signal initiating message sending;
d) sending the message to a peripheral unit;
e) sending a third signal to update states in a state repository.

[0038] In one embodiment, the method further comprises the step:

b2) sending a fourth signal, indicating that the message and the state has successfully been saved; and wherein the second signal of step c) is sent in response to receiving the fourth signal.

[0039] The above-stated method may further comprise features of the above-stated systems.

[0040] In one embodiment, the method further comprises the step:

f) receiving a fifth signal from the peripheral unit, indicating that the message has successfully been processed; wherein step e) is carried out in response to the message being successfully processed.

[0041] In particular, the object of the present invention is achieved by a computer-readable medium comprising logic for a gate array, the logic, when implemented in the gate array, causes the gate array to carry out the steps of one or several of the above stated methods or the above stated systems.

[0042] In particular, the object of the present invention is achieved by a computer-readable medium comprising instructions which, when executed by a computer, cause the computer to carry out the steps of one or several of the above stated methods or the above stated systems.

[0043] The benefits and advantages of the methods and the mediums are equal or similar to the advantages of the above-mentioned system.

[0044] In the following, embodiments of the invention are described with respect to the figures, wherein

Fig. 1 shows a flowchart for updating states relating to a message

Fig. 2 shows steps of a method for updating states

Fig. 3 shows a fail-operations system in a vehicle

Fig. 4 shows an embodiment of state storage

[0045] Fig. 1 shows a flowchart according to an embodiment of the inventions. The flowchart may contain a peripheral unit 102, an atom unit 103, a first execution unit 104, and a state keeper unit 106. The flowchart may optionally contain a decision unit 101 and a second execution unit 105.

[0046] The atom unit 103 may be configured to receive a first signal 110 from the first execution unit, indicating a message and a state. The atom unit 103 may further save the message and the state in the buffer (not shown). The atom unit may receive a second signal 120 from the first execution unit 104, requesting the initiation of send-

ing the message. The atom unit may, in response, send a signal 125 with the message to the peripheral unit 102 and send a third signal 135 to the state keeping unit 106 to update the states.

[0047] The atom unit 103 may further optionally send a fourth signal 115 to the first execution unit (104), after receiving the first signal 110, and after the message and the states have been stored, indicate that the message and the states has successfully been saved in the buffer. The atom unit may read the written information in the buffer and compare it with information received with the first signal 110 to determine that the message and states are stored correctly. The second signal 120 may be sent by the first execution unit 104 in response to receiving the fourth signal 115.

[0048] The atom unit 103 may further optionally receive a fifth signal 130 from the peripheral unit 102, indicating that the message has been processed successfully (such as sent on a network, preferably a CAN communication network, and/or received by another processing unit which responded with an operation successfully completed message). Updating the states may take place on reception from the peripheral unit, that the message has successfully been processed.

[0049] The decision unit may optionally receive a sixth signal 140 indicating a failure in the first execution unit 104. In response, the decision unit may disconnect the first execution unit 104 from the atom unit 103, and connect the second execution unit 105 with the atom unit 103 and/or the state keeping unit 106.

[0050] The second execution unit 105 may optionally, be initiated by the decision unit, and during initiation, read states from the atom unit and/or the state keeping unit. Although depicted as read from the state keeping unit 106 directly, the second execution unit may also read the states via the atom unit 103.

[0051] Fig. 2 shows a method according to an embodiment of the invention. The method may contain the steps: receiving 210 a first signal indicating a message and a state. Saving 220 the message and the state. In response to saving the message and the state, receiving 250 a second signal initiating message sending and sending 260 the message to a peripheral unit. The method may also comprise sending 280 a third signal to update states in a state repository.

[0052] Optionally, the method may contain a step sending 230 a fourth signal to the first execution unit, indicating that the message and the state has successfully been saved in the buffer and a step checking 240 if the message and the states are correctly saved in the buffer and if so, sending the fourth signal in response to checking the message and states are correctly saved in the buffer.

[0053] Optionally, the method may contain a step 270 receiving a fifth signal from the peripheral unit, indicating that the message has successfully been processed, and sending the third signal to update states, which is carried out in response to the signal indicating that the message

is being processed successfully.

[0054] Fig. 3 shows a system 310 for updating one or more states relating to a message in a vehicle. The system may comprise a peripheral unit 302, an atom unit 303, a first execution unit 304, and a state keeping unit 306. The system may also contain a second execution unit 305 and a decision unit 301.

[0055] The atom unit 303 may receive a first signal 110 from the first execution unit 304, indicating a message and states. The atom unit 303 may save the message and the states. The atom unit 303 may receive a second signal 120 from the first execution unit 304, requesting the initiation of sending the message and further sending the message to the peripheral unit 302 and updating the states in the state keeping unit 306.

[0056] Fig. 4 shows an embodiment where a buffer is part of the state keeping unit 306. In this embodiment, the state keeping unit 306 may comprise a first states storage 307, and a second states storage 308. One of the first and the second states storages may be identified as write states storage, and the other may be identified as read states storage. The read states storage may always be coherent with the messages sent to the peripheral unit. The atom unit may always write to the states storage identified as write state storage, such as when saving the message and the states in step b).

[0057] On receiving a second signal 120 to initiate the sending of the message and updating states, the identification of the write and the read states storage may switch, so that the one of the first and the second states storage that previously was the write states storage, becomes read states storage, and the other becomes write states storage. If a context switch takes place from the first execution unit 304 to the second execution unit 305, the second execution unit 305 may read from the one of the first states storage 307 and the second states storage 308 which is indicated as read state storage. This ensures, if later updates are made to the storage identified as write storage, the read storage will be consistent with the latest sent message.

[0058] After switching the read and write states storage, the new read storage may be copied to the new write storage to ensure that the new write storage is up to date. This may also be done after a context switch between primary and fail-operational components.

[0059] Logic may be implemented using gates, such as using a field-programmable gate array (FPGA) and executed simultaneously, or substantially simultaneously, such that the state storages switch substantially simultaneously with the message sent to the peripheral unit.

[0060] The one of the first state storage and the second state storage indicated as write state storage may in one embodiment also be read. This may be used to ensure that the states are correctly written to the write states storage before performing the read/write states storage switch. Other means to ensure integrity of states storage may be implemented depending on requirements, such

as parity bits or duplicating each states storage.

[0061] Reference numerals

101	decision unit
102	peripheral unit
103	atom unit
104	first execution unit
105	second execution unit
106	state keeping unit
110	first signal
115	fourth signal
120	second signal
125	sending message to peripheral unit
130	fifth signal
135	third signal
140	sixth signal
155	reading states from state keeping unit
210	receiving first signal
220	saving message and states
230	sending a fourth signal
240	checking saved message and states
250	receiving second signal
260	sending message to peripheral unit
270	receiving fifth signal
280	sending third signal
301	decision unit
302	peripheral unit
303	atom unit
304	first execution unit
305	second execution unit
306	state keeping unit
307	first states storage
308	second states storage

Claims

1. A system for updating one or more states relating to a message in a vehicle, the system comprising:

a first execution unit (104)
an atom unit (103);
a buffer;
a state keeping unit (106); and
a peripheral unit (102),

wherein the atom unit (103) is configured to carry out the steps:

- a) receiving a first signal (110) from the first execution unit (104), indicating the message and the states;
b) saving the message and the states in the buffer;
c) receiving a second signal (120) from the first execution unit (104), requesting the initiation of sending the message;
d) sending (125) the message to the peripheral

unit (102); and

e) sending a third signal (135) to the state keeping unit (106) to update the states.

2. The system of claim 1, wherein the atom unit (103) is operational independent of a failure and/or an error in the first execution unit.

3. The system of claim 2, wherein the failure and error is identified by inconsistency in lockstep execution or voting logic of the first execution unit (104), and/or the failure and error gives rise to an interrupt, preferably aborting the first execution unit operation.

4. The system of any of the preceding claims, wherein the atom unit (103) is further configured to carry out the step:
b2) sending a fourth signal (115) to the first execution unit (104), indicating that the message and the states have been successfully saved in the buffer; and wherein the second signal (120) of step c) is sent by the first execution unit (104) in response to receiving the fourth signal (115).

5. The system of claim 4, wherein the atom unit (103) is further configured to carry out the step:
b3) checking if the message and the states are correctly saved in the buffer, and sending the fourth signal (115) in response to checking the message and states are correctly saved in the buffer.

6. The system of any of the preceding claims, wherein the atom unit (103) is further configured to carry out the step:
f) receiving a fifth signal (130) from the peripheral unit (102), indicating that the message has been successfully processed;

wherein step e) is carried out when the message has been successfully processed.

7. The system of any of the preceding claims, wherein the atom unit (103) preferably comprises error correction means, such as voting logic.

8. The system of any of the preceding claims, further comprising:

a decision unit (101); and
a second execution unit (105);
wherein the decision unit is configured to:
in response to receiving a sixth signal (140) indicating a failure in the first execution unit (104),

disconnect the first execution unit (104) from the atom unit (103), and connect the second execution unit (105) with the atom unit (103) and/or the state keeping unit (106).

5

9. The system of claim 8, wherein, after the second execution unit is connected with the state keeping unit, the second execution unit reads (155) the states from the state keeping unit.

10

10. A computer-implemented method comprising steps:

a) receiving a first signal indicating a message and one or more states;
 b) saving the message and the states;
 c) in response to saving the message and the states, receiving a second signal initiating message sending;
 d) sending the message to a peripheral unit; and
 e) sending a third signal to update states in a state repository.

15

20

11. The method of claim 10, further comprising the step:
 b2) sending a fourth signal, indicating that the message and the states have been successfully saved in the buffer; and
 wherein the second signal of step c) is sent in response to receiving the fourth signal.

25

30

12. The method of claim 10 or claim 11, further comprising the step:
 f) receiving a fifth signal from the peripheral unit, indicating that the message has been successfully processed;
 wherein step e) is carried out in response to the message being successfully processed.

35

13. A computer-readable medium comprising logic for a gate array, the logic, when implemented in the gate array, causing the gate array to carry out the steps of any of the claims 10 to 12.

40

14. A computer-readable medium comprising instructions which, when executed by a computer, cause the computer to carry out the steps of the method of any of the claims 10 to 12.

45

50

55

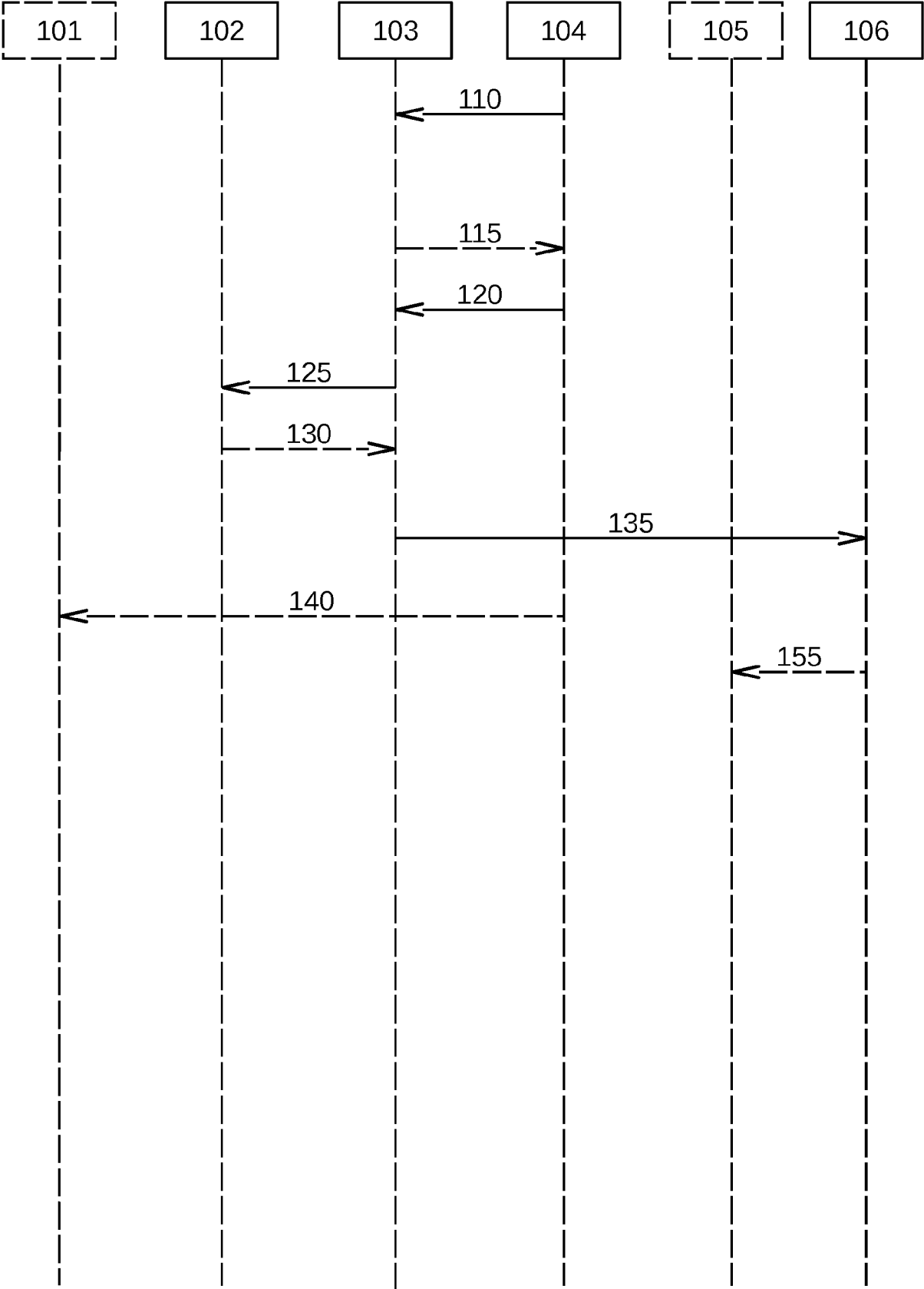


FIG 1

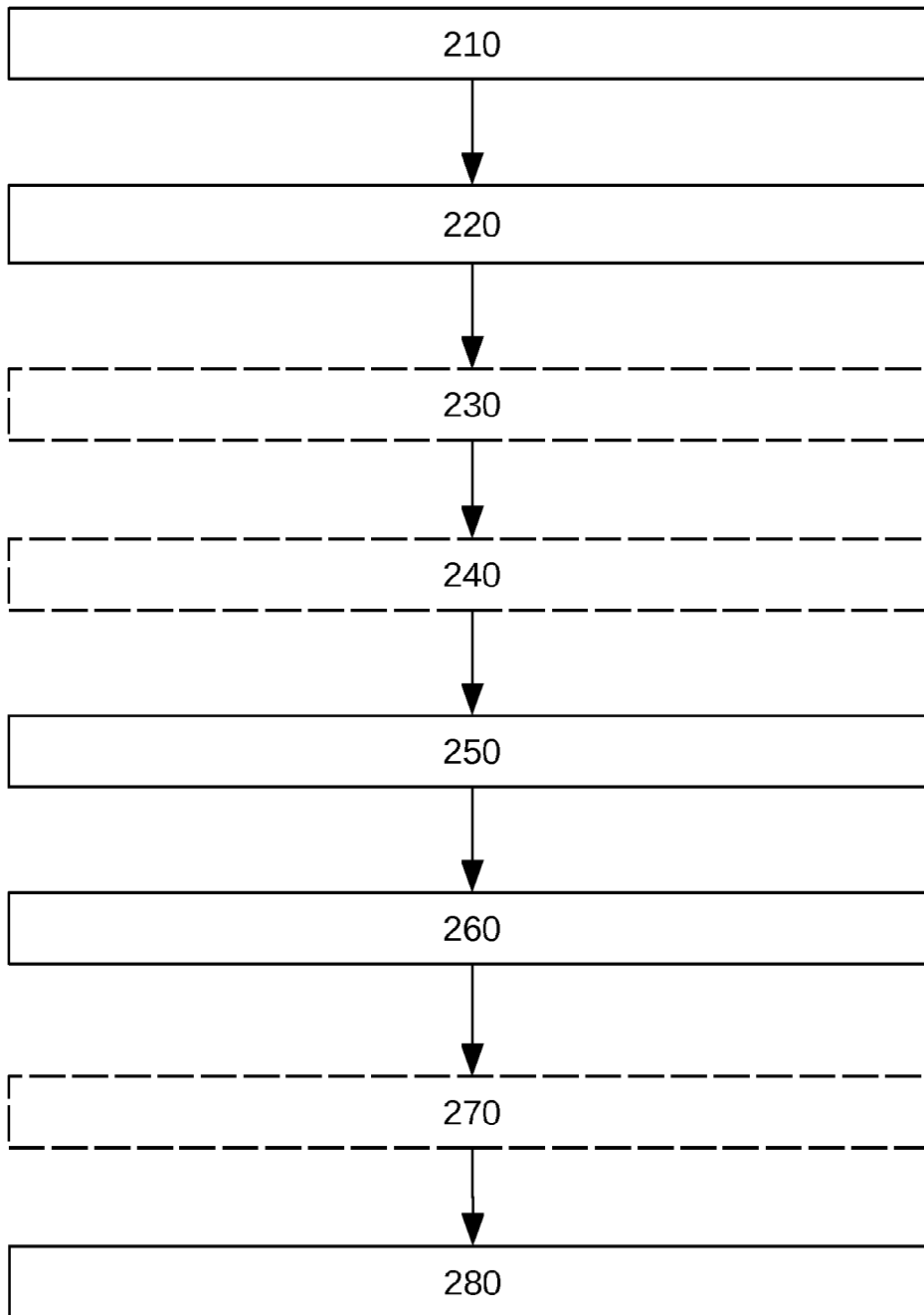


FIG 2

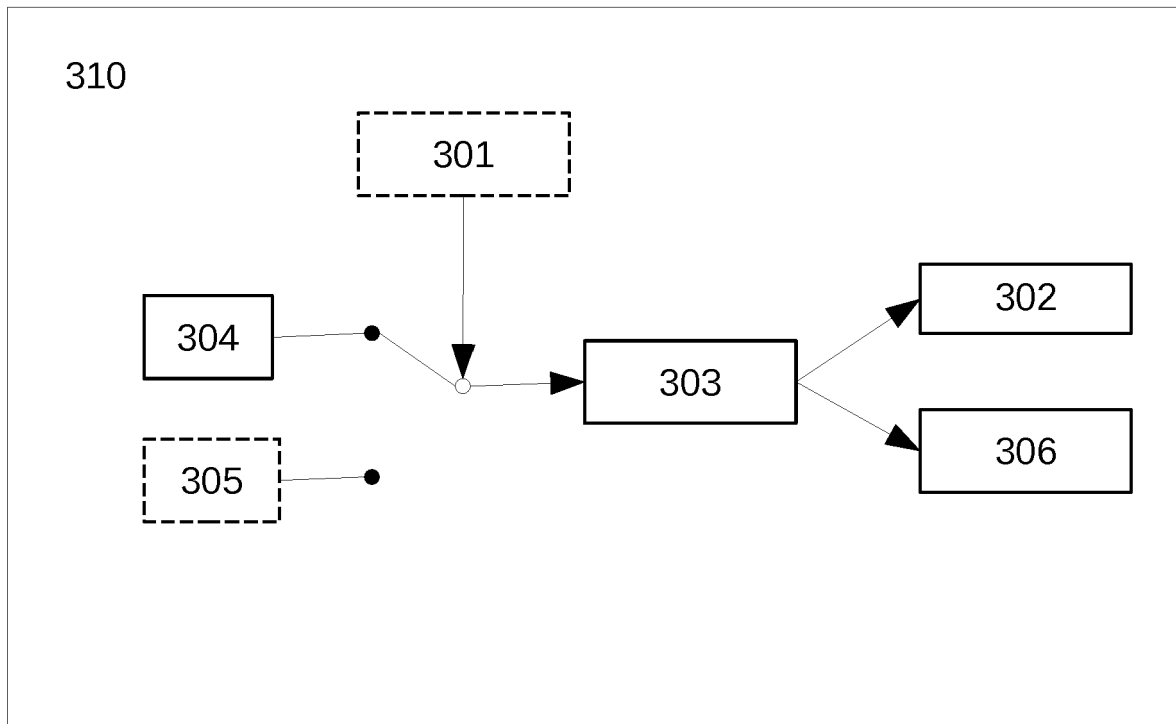


FIG 3

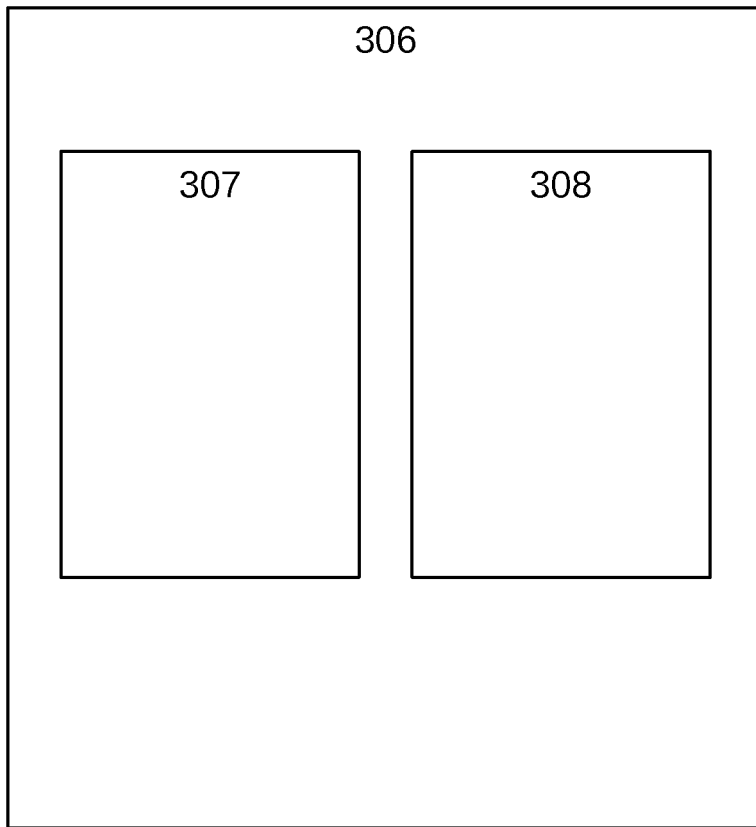


FIG 4



EUROPEAN SEARCH REPORT

Application Number
EP 19 17 0019

5

10

15

20

25

30

35

40

45

50

55

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
A	US 2005/132250 A1 (HANSEN ROGER [US] ET AL) 16 June 2005 (2005-06-16) * abstract; figures 1a-1c * * paragraphs [0008], [0022], [0033] - [0038] *	1-14	INV. G06F11/07 G06F11/14 G06F11/16 G06F11/18 G06F11/20
A	WO 01/46806 A1 (INTEL CORP [US]; QUACH NHON TOAI [US]) 28 June 2001 (2001-06-28) * abstract * * figures 3,4,5a,5b * * page 12, line 23 - page 16, line 22 *	1-14	
A	US 2018/032413 A1 (NAGANATHAN VINOD SHANKAR [US]) 1 February 2018 (2018-02-01) * paragraphs [0005], [0006] *	1-14	
A	US 8 443 230 B1 (JAMES-ROXBY PHILIP B [US] ET AL) 14 May 2013 (2013-05-14) * abstract *	1-14	
			TECHNICAL FIELDS SEARCHED (IPC)
			G06F
The present search report has been drawn up for all claims			
Place of search Munich		Date of completion of the search 16 October 2019	Examiner Weber, Vincent
<p>CATEGORY OF CITED DOCUMENTS</p> <p>X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document</p> <p>T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document</p>			

 1
EPO FORM 1503 03.82 (P04C01)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 19 17 0019

5 This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

16-10-2019

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2005132250 A1	16-06-2005	DE 102004038649 A1	21-07-2005
		JP 2005182781 A	07-07-2005
		US 2005132250 A1	16-06-2005
WO 0146806 A1	28-06-2001	AU 1631801 A	03-07-2001
		CN 1434941 A	06-08-2003
		CN 101539875 A	23-09-2009
		DE 10085324 T1	05-12-2002
		GB 2373900 A	02-10-2002
		HK 1046573 A1	06-05-2005
		TW I247986 B	21-01-2006
		US 6625749 B1	23-09-2003
		US 2004019771 A1	29-01-2004
		WO 0146806 A1	28-06-2001
US 2018032413 A1	01-02-2018	CN 107666423 A	06-02-2018
		DE 102017116883 A1	01-02-2018
		US 2018032413 A1	01-02-2018
US 8443230 B1	14-05-2013	NONE	